TASK 5

To analyze traffic accident data to identify patterns related to road conditions,weather, and time of day and visualize accident hotspots and contributing factors.

```
#Import required libraries
import pandas as pd
import numpy as np
#import seaborn as sns
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure
import seaborn as sns
```

```
#Load the data
Data=pd.read_csv('/content/drive/MyDrive/UK_Accident.csv')
Data

#Create a dataframe
df=pd.DataFrame(Data)
df
```

	Unnamed: 0	Accident_Index	Location_Easting_OSGR	Location_Northing_OSGR	Lor
0	0	200501BS00001	525680.0	178240.0	-0
1	1	200501BS00002	524170.0	181650.0	-0
2	2	200501BS00003	524520.0	182240.0	-0
3	3	200501BS00004	526900.0	177530.0	-0
4	4	200501BS00005	528060.0	179040.0	-0
...	...	...	...	...	
1504145	464692	2.01E+12	310037.0	597647.0	-3
1504146	464693	2.01E+12	321509.0	574063.0	-3
1504147	464694	2.01E+12	321337.0	566365.0	-3
1504148	464695	2.01E+12	323869.0	566853.0	-3
1504149	464696	2.01E+12	314072.0	579971.0	-3

1504150 rows × 33 columns

```
#First 5 rows of the dataset
df.head()
```

	Unnamed: 0	Accident_Index	Location_Easting_OSGR	Location_Northing_OSGR	Longitude
0	0	200501BS00001	525680.0	178240.0	-0.191170
1	1	200501BS00002	524170.0	181650.0	-0.211708

```
#Last 5 rows of the dataset
df.tail()
```

	Unnamed: 0	Accident_Index	Location_Easting_OSGR	Location_Northing_OSGR	Lor
1504145	464692	2.01E+12	310037.0	597647.0	-3
1504146	464693	2.01E+12	321509.0	574063.0	-3
1504147	464694	2.01E+12	321337.0	566365.0	-3
1504148	464695	2.01E+12	323869.0	566853.0	-3
1504149	464696	2.01E+12	314072.0	579971.0	-3

5 rows × 33 columns

```
#Check for all the columns of the dataset
df.columns
```

```
Index(['Unnamed: 0', 'Accident_Index', 'Location_Easting_OSGR',
       'Location_Northing_OSGR', 'Longitude', 'Latitude', 'Police_Force',
       'Accident_Severity', 'Number_of_Vehicles', 'Number_of_Casualties',
       'Date', 'Day_of_Week', 'Time', 'Local_Authority_(District)',
       'Local_Authority_(Highway)', '1st_Road_Class', '1st_Road_Number',
       'Road_Type', 'Speed_limit', 'Junction_Control', '2nd_Road_Class',
       '2nd_Road_Number', 'Pedestrian_Crossing-Human_Control',
       'Pedestrian_Crossing-Physical_Facilities', 'Light_Conditions',
       'Weather_Conditions', 'Road_Surface_Conditions',
       'Special_Conditions_at_Site', 'Carriageway_Hazards',
       'Urban_or_Rural_Area', 'Did_Police_Officer_Attend_Scene_of_Accident',
       'LSOA_of_Accident_Location', 'Year'],
      dtype='object')
```

```
#Check for the number of rows and columns of the dataset
df.shape
```

```
(1504150, 33)
```

```
#Check for the information ,i.e, dtype and null value for each column
df.info
```

```
1504145                 None                   2
1504146                 None                   2
1504147                 None                   2
1504148                 None                   2
1504149                 None                   2

        Did_Police_Officer_Attend_Scene_of_Accident  \
0                                               Yes   
1                                               Yes   
2                                               Yes   
3                                               Yes   
4                                               Yes   
...                                             ...   
1504145                                         Yes   
1504146                                         Yes   
1504147                                         Yes   
1504148                                         Yes   
1504149                                         Yes   

         LSOA_of_Accident_Location  Year  
0                        E01002849  2005  
1                        E01002909  2005  
2                        E01002857  2005  
3                        E01002840  2005  
4                        E01002863  2005  
...                            ...   ...  
1504145                        NaN  2014  
1504146                        NaN  2014  
1504147                        NaN  2014  
1504148                        NaN  2014  
1504149                        NaN  2014  

[1504150 rows x 33 columns]>
```

```
#Check for Statistical Analysis
df.describe
```

```
    1504147                        NaN  2014
    1504148                        NaN  2014
    1504149                        NaN  2014

    [1504150 rows x 33 columns]>
```

```
#Datatype of each column
df.dtypes
```

```
    Unnamed: 0                                       int64
    Accident_Index                                  object
    Location_Easting_OSGR                          float64
    Location_Northing_OSGR                         float64
    Longitude                                      float64
    Latitude                                       float64
    Police_Force                                     int64
    Accident_Severity                                int64
    Number_of_Vehicles                               int64
    Number_of_Casualties                             int64
    Date                                            object
    Day_of_Week                                      int64
    Time                                            object
    Local_Authority_(District)                       int64
    Local_Authority_(Highway)                       object
    1st_Road_Class                                   int64
    1st_Road_Number                                  int64
    Road_Type                                       object
    Speed_limit                                      int64
    Junction_Control                                object
    2nd_Road_Class                                   int64
    2nd_Road_Number                                  int64
    Pedestrian_Crossing-Human_Control               object
    Pedestrian_Crossing-Physical_Facilities         object
    Light_Conditions                                object
    Weather_Conditions                              object
    Road_Surface_Conditions                         object
    Special_Conditions_at_Site                      object
    Carriageway_Hazards                             object
    Urban_or_Rural_Area                              int64
    Did_Police_Officer_Attend_Scene_of_Accident     object
    LSOA_of_Accident_Location                       object
    Year                                             int64
    dtype: object
```

```
#  DATA CLEANING
```

```
#Check for the null values
print(df.isnull().sum())
```

```
    Unnamed: 0                                          0
    Accident_Index                                      0
    Location_Easting_OSGR                             101
    Location_Northing_OSGR                              0
    Longitude                                         101
    Latitude                                            0
    Police_Force                                        0
    Accident_Severity                                   0
    Number_of_Vehicles                                  0
    Number_of_Casualties                                0
    Date                                                0
    Day_of_Week                                         0
    Time                                              117
    Local_Authority_(District)                          0
    Local_Authority_(Highway)                           0
    1st_Road_Class                                      0
    1st_Road_Number                                     0
    Road_Type                                           0
    Speed_limit                                         0
    Junction_Control                                    0
    2nd_Road_Class                                      0
    2nd_Road_Number                                     0
    Pedestrian_Crossing-Human_Control                  17
    Pedestrian_Crossing-Physical_Facilities            34
    Light_Conditions                                    0
    Weather_Conditions                                  0
    Road_Surface_Conditions                             0
    Special_Conditions_at_Site                          0
    Carriageway_Hazards                                 0
    Urban_or_Rural_Area                                 0
    Did_Police_Officer_Attend_Scene_of_Accident         0
    LSOA_of_Accident_Location                      108238
    Year                                                0
    dtype: int64
```

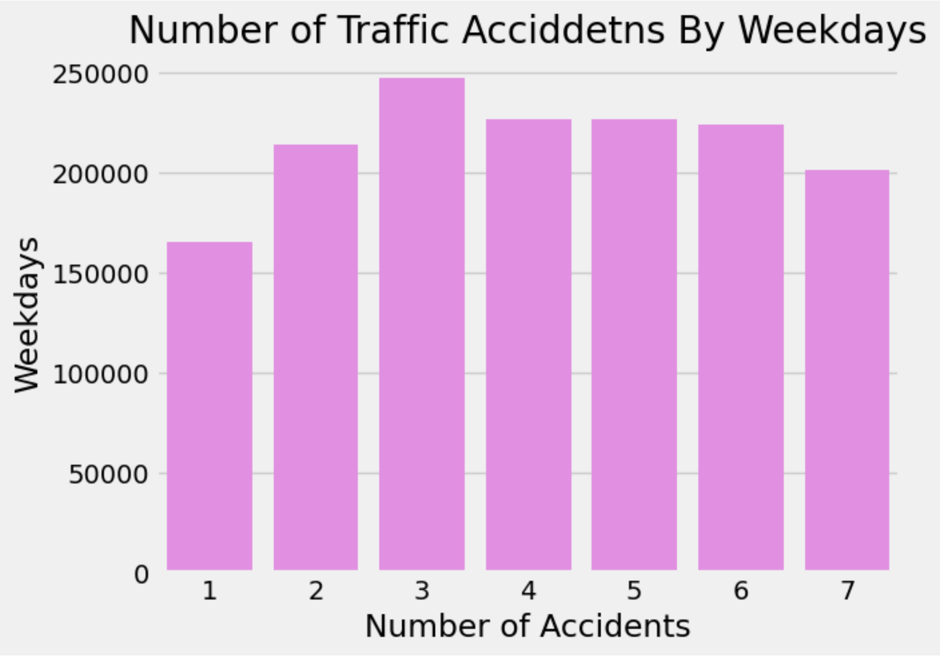
```
#Remove unnecesary columns
df1=df.drop(['Location_Easting_OSGR','Longitude','LSOA_of_Accident_Location','Pedestrian_Crossing-Human_Control','Pedestrian_Crossing-Ph
df1

#New dataframe without unnecesary columns
print(df1.isnull().sum())
```

```
Unnamed: 0                                       0
Accident_Index                                   0
Location_Northing_OSGR                           0
Latitude                                         0
Police_Force                                     0
Accident_Severity                                0
Number_of_Vehicles                               0
Number_of_Casualties                             0
Date                                             0
Day_of_Week                                      0
Time                                           117
Local_Authority_(District)                       0
Local_Authority_(Highway)                        0
1st_Road_Class                                   0
1st_Road_Number                                  0
Road_Type                                        0
Speed_limit                                      0
Junction_Control                                 0
2nd_Road_Class                                   0
2nd_Road_Number                                  0
Light_Conditions                                 0
Weather_Conditions                               0
Road_Surface_Conditions                          0
Special_Conditions_at_Site                       0
Carriageway_Hazards                              0
Urban_or_Rural_Area                              0
Did_Police_Officer_Attend_Scene_of_Accident      0
Year                                             0
dtype: int64
```

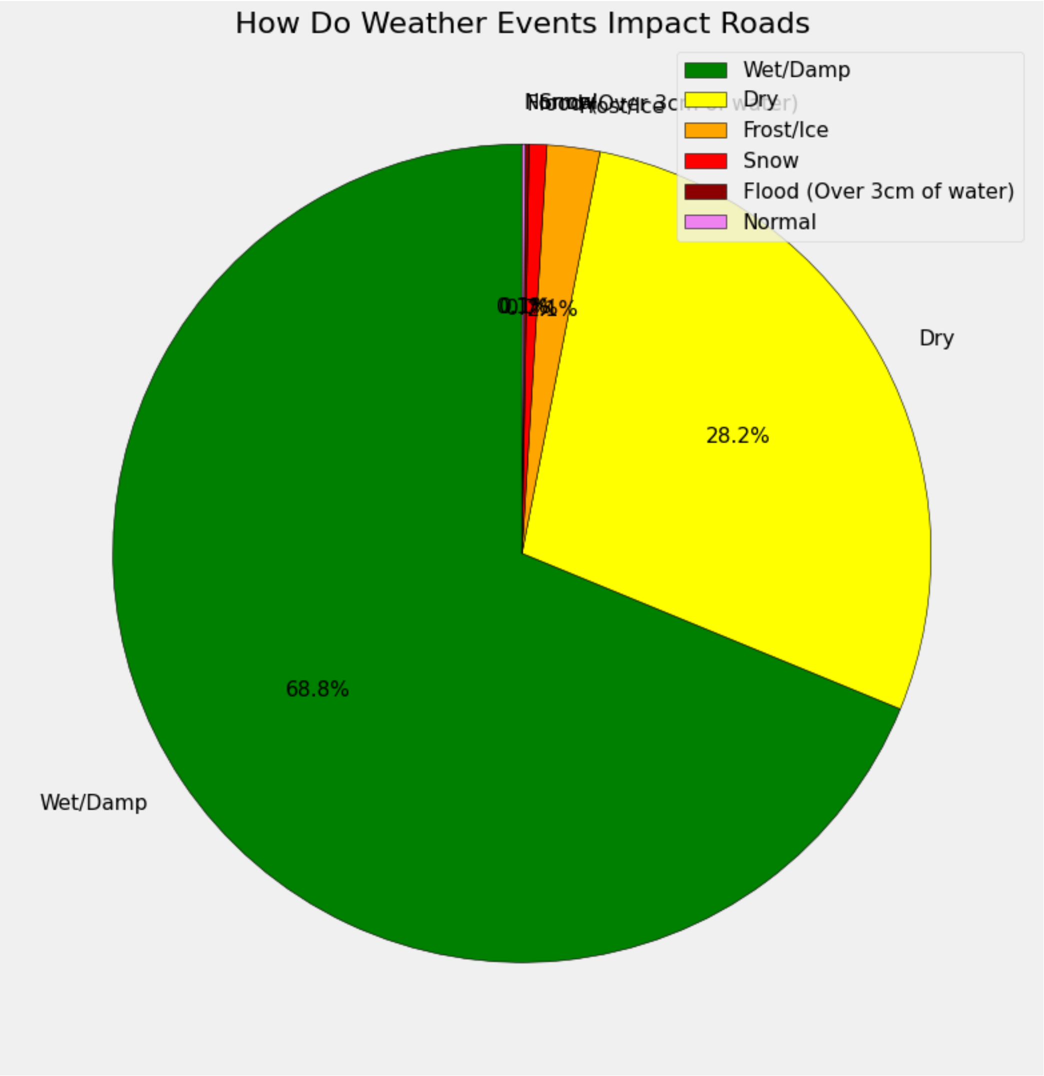
DATA VISUALIZATION

```
#Accident rates by week
week = df1["Day_of_Week"].value_counts()
week
weeks = df1["Day_of_Week"].unique()
num_weeks = week.values
sns.barplot(x=weeks,y=num_weeks,color='violet')
plt.title("Number of Traffic Acciddetns By Weekdays")
plt.xlabel("Number of Accidents")
plt.ylabel("Weekdays")
```

```
Text(0, 0.5, 'Weekdays')
```

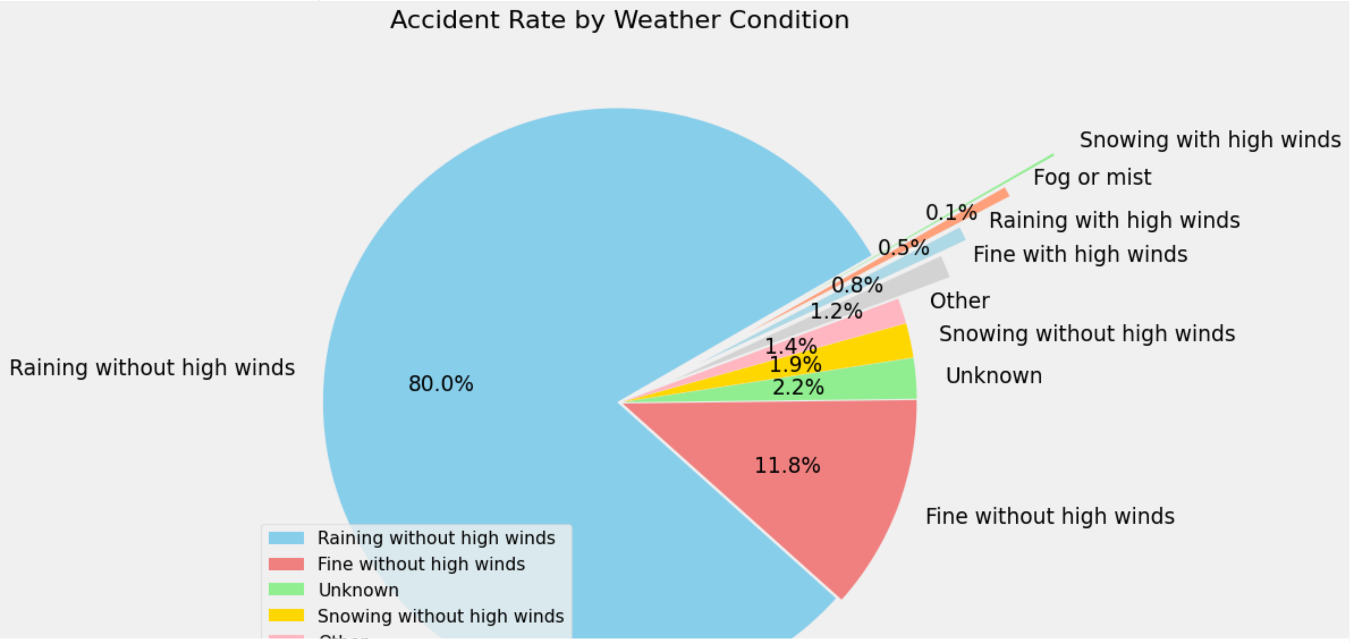


```
#Accident rates due to road conditions
road_conditions = df1["Road_Surface_Conditions"].value_counts()
road_conditions_values = df1["Road_Surface_Conditions"].unique()
custom___colors = ['green', 'yellow', 'orange', 'red', 'darkred','violet']

figure(figsize=(10, 10), dpi=80)
plt.pie(road_conditions,labels = road_conditions_values,autopct="%1.1f%%",wedgeprops={'edgecolor':'black'},
        startangle=90,colors=custom___colors)
plt.tight_layout()
plt.legend()
plt.title("How Do Weather Events Impact Roads")
plt.show()
```



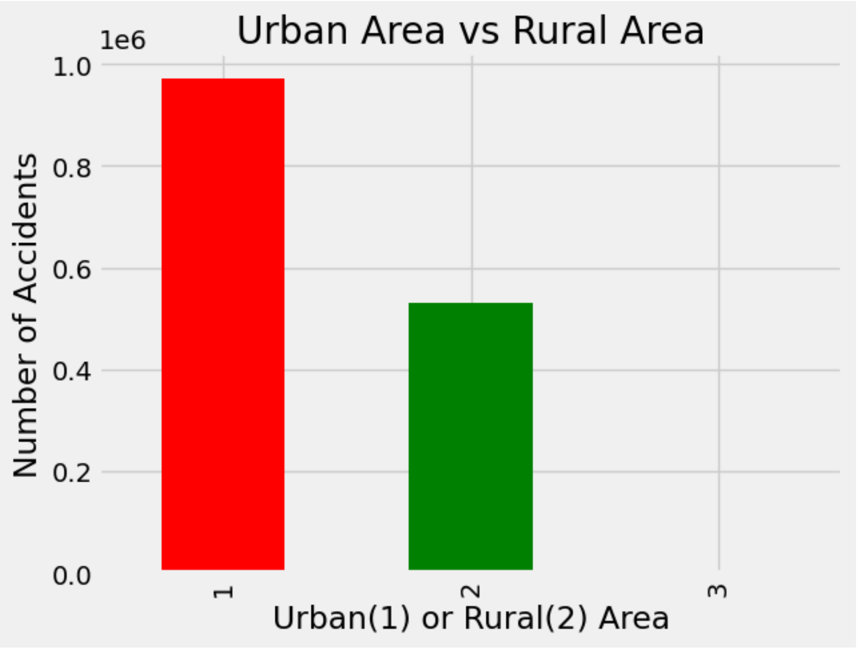
```
#Accident rates due to weather conditions
weather_cond = df1["Weather_Conditions"].value_counts()
weather_cond_values= df1["Weather_Conditions"].unique()
weather_num_acc_arr = weather_cond.values
custom_colors_ = ['skyblue', 'lightcoral', 'lightgreen', 'gold', 'lightpink', 'lightgrey', 'lightblue', 'lightsalmon', 'lightgreen']
figure(figsize=(10, 10), dpi=80)
plt.pie(weather_cond, labels = weather_cond_values,startangle = 30,textprops={'size': 'large'},explode=(0.01,0.01,0.01,0.01,0.01,0.20,0.3
plt.legend(loc ="lower left")
plt.title("Accident Rate by Weather Condition")
```

```
Text(0.5, 1.0, 'Accident Rate by Weather Condition')
```

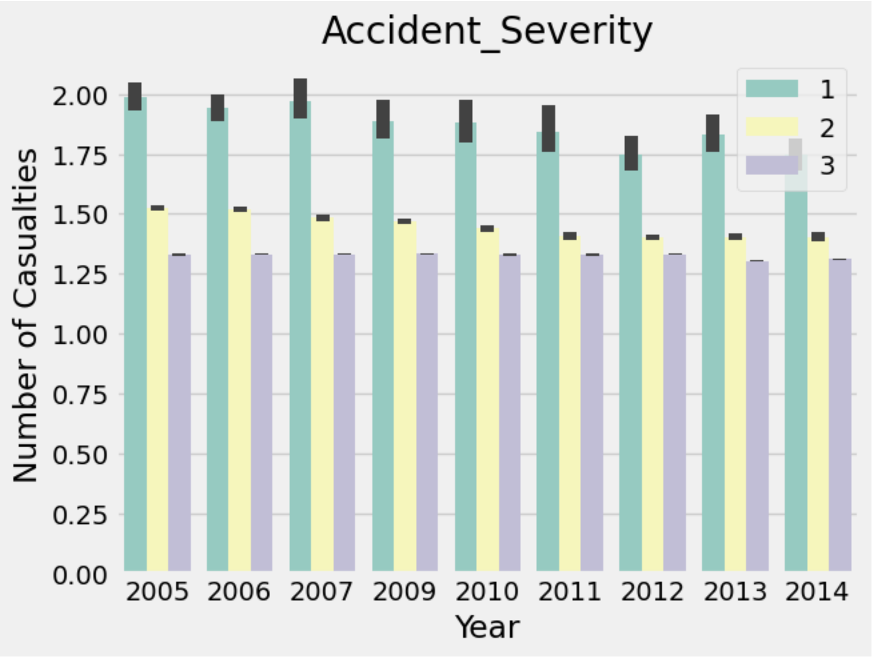


```
#Accidents rates in urban and rural areas
plt.style.use('fivethirtyeight')
df1["Urban_or_Rural_Area"].value_counts().plot(kind='bar', color=['red', 'green','yellow'])
plt.ylabel("Number of Accidents")
plt.xlabel("Urban(1) or Rural(2) Area")
plt.title("Urban Area vs Rural Area")
```

```
Text(0.5, 1.0, 'Urban Area vs Rural Area')
```



```
#Number of injured in accidents and the severity of accidents
sns.barplot(x="Year",y="Number_of_Casualties",data=df1,hue="Accident_Severity",palette="Set3")
plt.title("Accident_Severity")
plt.ylabel("Number of Casualties")
plt.legend(loc="upper right")
plt.show()
```


Accident_Severity
Number of Casualties
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
2005
2006
2007
2009
2010
2011
2012
2013
2014
Year
1
2
3